

**Amendments to the Claims**

Claims 1 - 6 (canceled)

1 Claim 7 (currently amended): A computer program product for improving performance and  
2 resource utilization of software applications that interact with a back-end data source to update  
3 information stored therein, the computer program product embodied on one or more computer-  
4 readable media and comprising:

5 computer-readable program code means for storing one or more objects in a cache for  
6 responding to update requests against a back-end data source corresponding to each of the objects,  
7 wherein (1) a set of input properties is stored with or associated with each stored object and (2)  
8 update logic specifying how to update the back-end data source corresponding to each of the stored  
9 objects is stored with or associated with the stored object or a group of stored objects;

10 computer-readable program code means for receiving update requests against one or more of  
11 the objects;

12 computer-readable program code means for checking an update policy to determine  
13 determining an update mode to use for a selected update request, responsive to the computer-  
14 readable program code means for receiving;

15 computer-readable program code means for immediately processing the selected update  
16 request against the back-end data source if the determined update mode is not a delayed update  
17 mode; and

18 computer-readable program code means for delaying processing of the selected update  
19 request against the back-end data source otherwise.

Serial No. 09/611,157

-4-

Docket RSW9-2000-0034-US1

1 Claim 8 (original): The computer program product according to Claim 7, wherein the computer-  
2 readable program code means for delaying processing further comprises:

3 computer-readable program code means for queuing the selected update request, along with  
4 the input properties and values thereof which are to be used for performing the selected update  
5 request, as a queued update request on an update queue;

6 computer-readable program code means for detecting a triggering event for performing the  
7 delayed processing of the queued update requests; and

8 computer-readable program code means for performing, responsive to the computer-readable  
9 program code means for detecting, the queued update requests.

1 Claim 9 (currently amended): The computer program product according to Claim 8, wherein the  
2 computer-readable program code means for performing further comprises:

3 computer-readable program code means for setting the input properties of a selected object  
4 against which the queued update request is to be performed using the queued input property values;  
5 and

6 computer-readable program code means for executing the update logic stored with or  
7 associated with the selected object, thereby updating the back-end data source corresponding to the  
8 selected object.

1 Claim 10 (original): The computer program product according to Claim 8, wherein the triggering  
2 event comprises reaching a particular count of queued update requests for a selected object.

Serial No. 09/611,157

-5-

Docket RSW9-2000-0034-US1

1 Claim 11 (original): The computer program product according to Claim 8, wherein the triggering  
2 event comprises reaching a particular time of day.

1 Claim 12 (currently amended): The computer program product according to Claim 8, wherein the  
2 triggering event comprises a count of read requests received for information about an associated  
3 object which is used for responding to read requests.

1 Claim 13 (original): The computer program product according to Claim 8, wherein a separate  
2 update queue is created for each of one or more back-end data sources to be accessed during  
3 operation of the computer-readable program code means for performing.

1 Claim 14 (original): The computer program product according to Claim 7, wherein the computer-  
2 readable program code means for determining further comprises computer-readable program code  
3 means for selecting the delayed update mode based upon a time of day when the selected update  
4 request is received.

1 Claim 15 (original): The computer program product according to Claim 7, wherein the computer-  
2 readable program code means for determining further comprises computer-readable program code  
3 means for selecting the delayed update mode based upon a classification of a user making the  
4 selected update request.

1 Claim 16 (original): The computer program product according to Claim 8, further comprising:  
2 computer-readable program code means for connecting to the back-end data source prior to  
3 operation of the computer-readable program code means for performing; and  
4 computer-readable program code means for disconnecting from the back-end data source  
5 after operation of the computer-readable program code means for performing.

Claims 17 - 22 (canceled)

1 Claim 23 (currently amended): A system for improving performance and resource utilization of  
2 software applications that interact with a back-end data source to update information stored therein,  
3 comprising:  
4 means for storing one or more objects in a cache for responding to update requests against a  
5 back-end data source corresponding to each of the objects, wherein (1) a set of input properties is  
6 stored with or associated with each stored object and (2) update logic specifying how to update the  
7 back-end data source corresponding to each of the stored objects is stored with or associated with  
8 the stored object or a group of stored objects;  
9 means for receiving update requests against one or more of the objects;  
10 means for ~~determining~~ checking an update policy to determine an update mode to use for a  
11 selected update request, responsive to the means for receiving;  
12 means for immediately processing the selected update request against the back-end data  
13 source if the determined update mode is not a delayed update mode; and  
14 means for delaying processing of the selected update request against the back-end data

Serial No. 09/611,157

-7-

Docket RSW9-2000-0034-US1

15 source otherwise.

1 Claim 24 (original): The system according to Claim 23, wherein the means for delaying processing  
2 further comprises:

3 means for queuing the selected update request, along with the input properties and values  
4 thereof which are to be used for performing the selected update request, as a queued update request  
5 on an update queue;

6 means for detecting a triggering event for performing the delayed processing of the queued  
7 update requests; and

8 means for performing, responsive to the means for detecting, the queued update requests.

1 Claim 25 (currently amended): The system according to Claim 24, wherein the means for  
2 performing further comprises:

3 means for setting the input properties of a selected object against which the queued update  
4 request is to be performed using the queued input property values; and

5 means for executing the update logic stored with or associated with the selected object,  
6 thereby updating the back-end data source corresponding to the selected object.

1 Claim 26 (original): The system according to Claim 24, wherein the triggering event comprises  
2 reaching a particular count of queued update requests for a selected object.

1 Claim 27 (original): The system according to Claim 24, wherein the triggering event comprises

2 reaching a particular time of day.

1 Claim 28 (currently amended): The system according to Claim 24, wherein the triggering event  
2 ~~comprises information about a count of read requests received for~~ an associated object which is used  
3 for responding to read requests.

1 Claim 29 (original): The system according to Claim 24, wherein a separate update queue is created  
2 for each of one or more back-end data sources to be accessed during operation of the means for  
3 performing.

1 Claim 30 (original): The system according to Claim 23, wherein the means for determining further  
2 comprises means for selecting the delayed update mode based upon a time of day when the selected  
3 update request is received.

1 Claim 31 (original): The system according to Claim 23, wherein the means for determining further  
2 comprises means for selecting the delayed update mode based upon a classification of a user making  
3 the selected update request.

1 Claim 32 (original): The system according to Claim 24, further comprising:  
2 means for connecting to the back-end data source prior to operation of the means for  
3 performing; and  
4 means for disconnecting from the back-end data source after operation of the means for

5 performing.

Claims 33 - 38 (canceled)

1 Claim 39 (currently amended): A method for improving performance and resource utilization of  
2 software applications that interact with a back-end data source to update information stored therein,  
3 comprising the steps of:

4 storing one or more objects in a cache for responding to update requests against a back-end  
5 data source corresponding to each of the objects, wherein (1) a set of input properties is stored with  
6 or associated with each stored object and (2) update logic specifying how to update the back-end  
7 data source corresponding to each of the stored objects is stored with or associated with the stored  
8 object or a group of stored objects;

9 receiving update requests against one or more of the objects;

10 checking an update policy to determine determining an update mode to use for a selected  
11 update request, responsive to the receiving step;

12 immediately processing the selected update request against the back-end data source if the  
13 determined update mode is not a delayed update mode; and

14 delaying processing of the selected update request against the back-end data source  
15 otherwise.

1 Claim 40 (original): The method according to Claim 39, wherein the step of delaying processing  
2 further comprises the steps of:

Serial No. 09/611,157

-10-

Docket RSW9-2000-0034-US1

3 queuing the selected update request, along with the input properties and values thereof which  
4 are to be used for performing the selected update request, as a queued update request on an update  
5 queue;

6 detecting a triggering event for performing the delayed processing of the queued update  
7 requests; and

8 performing, responsive to the detecting step, the queued update requests.

1 Claim 41 (currently amended): The method according to Claim 40, wherein the performing step  
2 further comprises the steps of:

3 setting the input properties of a selected object against which the queued update request is to  
4 be performed using the queued input property values; and

5 executing the update logic stored with or associated with the selected object, thereby  
6 updating the back-end data source corresponding to the selected object.

1 Claim 42 (original): The method according to Claim 40, wherein the triggering event comprises  
2 reaching a particular count of queued update requests for a selected object.

1 Claim 43 (original): The method according to Claim 40, wherein the triggering event comprises  
2 reaching a particular time of day.

1 Claim 44 (currently amended): The method according to Claim 40, wherein the triggering event  
2 comprises ~~information about~~ a count of read requests received for an associated object which is used



3 for responding to read requests.

1 Claim 45 (original): The method according to Claim 40, wherein a separate update queue is created  
2 for each of one or more back-end data sources to be accessed during operation of the step of  
3 performing.

1 Claim 46 (original): The method according to Claim 39, wherein the determining step further  
2 comprises the step of selecting the delayed update mode based upon a time of day when the selected  
3 update request is received.

1 Claim 47 (original): The method according to Claim 39, wherein the determining step further  
2 comprises the step of selecting the delayed update mode based upon a classification of a user  
3 making the selected update request.

1 Claim 48 (original): The method according to Claim 40, further comprising the steps of:  
2 connecting to the back-end data source prior to operation of the performing step; and  
3 disconnecting from the back-end data source after operation of the performing step.